Recent MDPv2 Congestion Control Research Status

Brian Adamson / Joe Macker

5 June 1999

MDPv2 Protocol Overview

- MDPv2 is a selective, negative acknowledgement (NACK) reliability protocol which uses hybrid FEC-based repair.
- MDPv2 is well-suited for one-to-many and many-to-many bulk file/data transfer and is extensible to support additional multicast session paradigms.
- MDPv2 nodes collect round-trip timing information to dynamically set repair cycle and NACK suppression timers to adapt to different network environments.
- Special features to support operation across a variety of network architectures (e.g. wireless or asymmetric topologies) are also included.

Congestion Control Approach

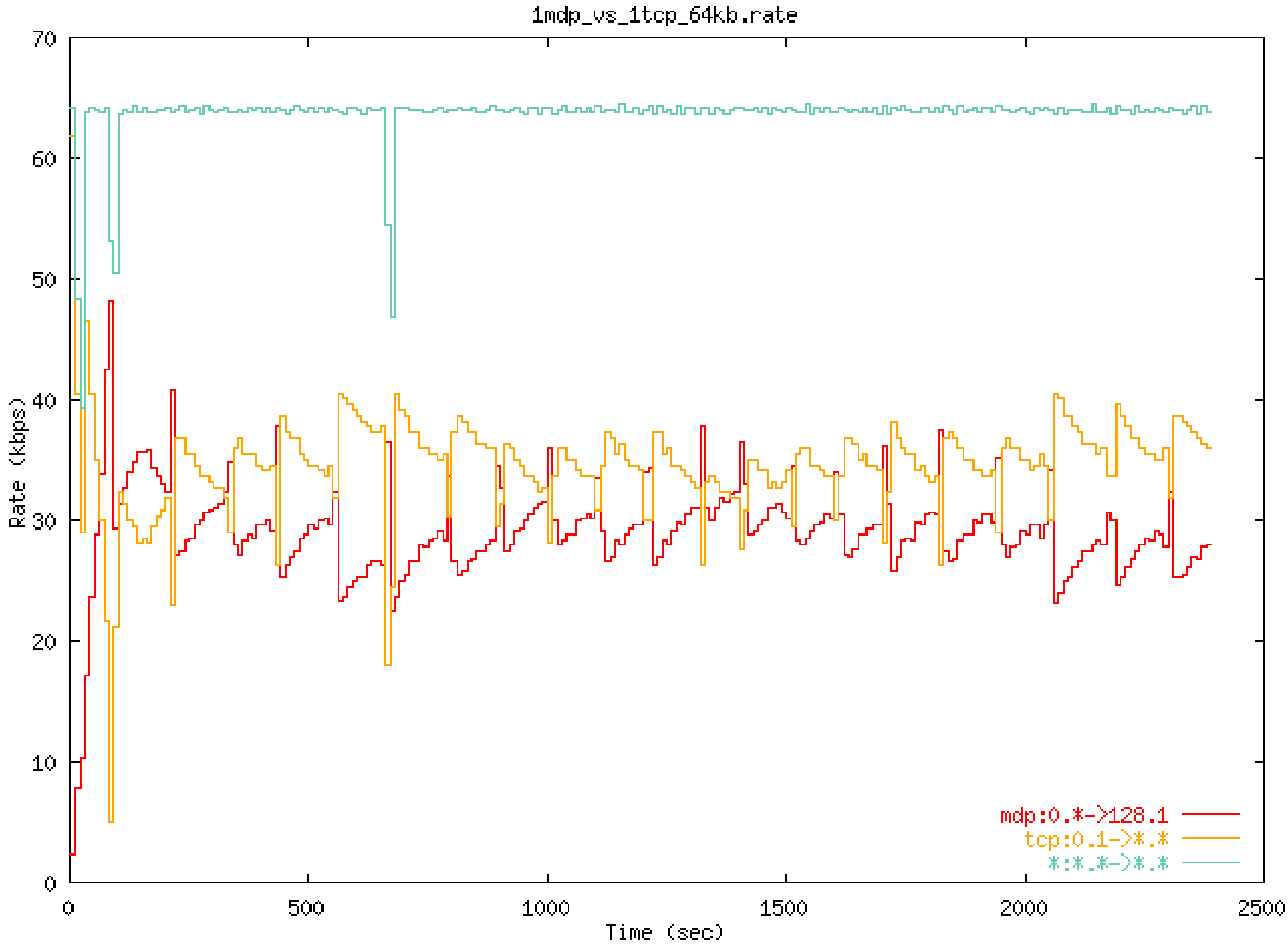
- The MDPv2 protocol has been extended to include optional experimental congestion control features loosely based on the Dec 1998 IRTF RMG “Strawman” specification.
- MDPv2 receivers (clients) report packet loss estimates in addition to round-trip delay in NACK messages _and_ in ACK responses to “probes” generated by data transmission sources (servers).
- The server(s) use these loss and delay estimates to determine a suitable transmission rate based on the UMass TCP throughput model.
- A small subset of “designated receivers” (or representatives) is elected for rapid probing and the “worst path” node among these bottleneck candidates is used to determine the current transmit rate.
- The election of “designated receivers” is a dynamic and continuous process based upon updates of loss and roundtrip time estimates.
- The current approach preserves much of the NACK-based nature of the protocol and is intended to scale to large group sizes for bulk transfer applications.

Congestion Control Approach (cont’d)

- Noted differences from “Strawman” recommendation:
  - Server collects, maintains, and advertises round-trip estimates (I.e. client have no knowledge of their individual round-trip estimates). This approach runs with low overhead and is adaptable to asymmetric network topologies.
  - Slightly different packet loss estimation algorithm is used. A type of exponentially weighted moving average is currently used. Loss estimation techniques are under further investigation.
  - Use of rapid probing of “designated receiver” nodes to closely track “worst path” bottleneck dynamics.
  - Some congestion control dynamics use presence/lack of “designated receiver” feedback to adjust transmission rate. This improves performance under low bandwidth and/or large round-trip topologies.

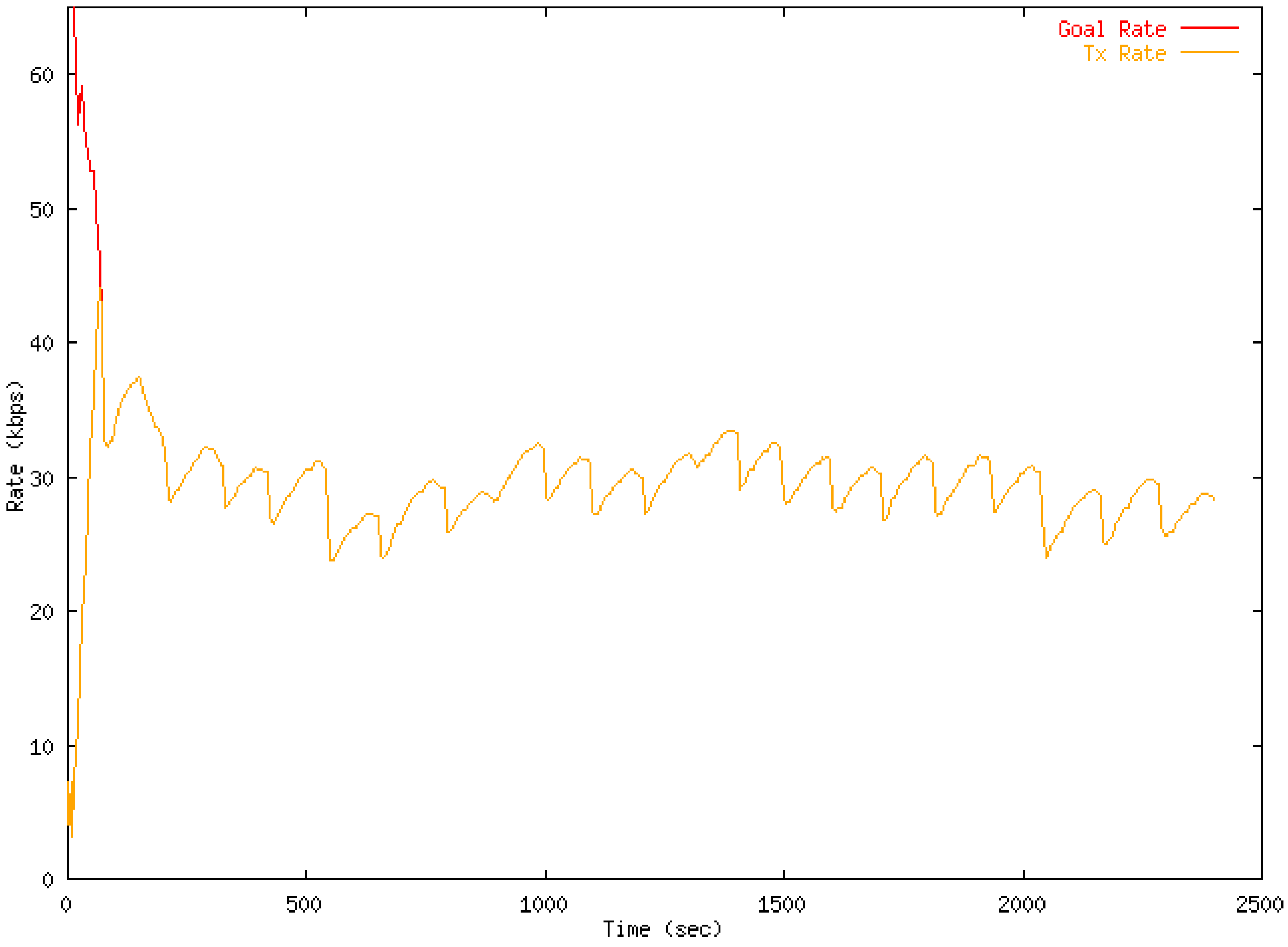
Initial Results

- MDPv2 has been ported into ns-2. The same protocol engine code is used in the simulation environment and real applications.
- We have measured MDP vs. TCP on simple multicast topologies with bottleneck links in the *ns-2* simulator and on real network connections.
- The following are graphs of some preliminary results collected in *ns-2*.
- The initial results validate the steady-state applicability of the UMass model in a real protocol implementation. Further work is needed to evaluate the technique with additional network dynamics and more complex topologies.

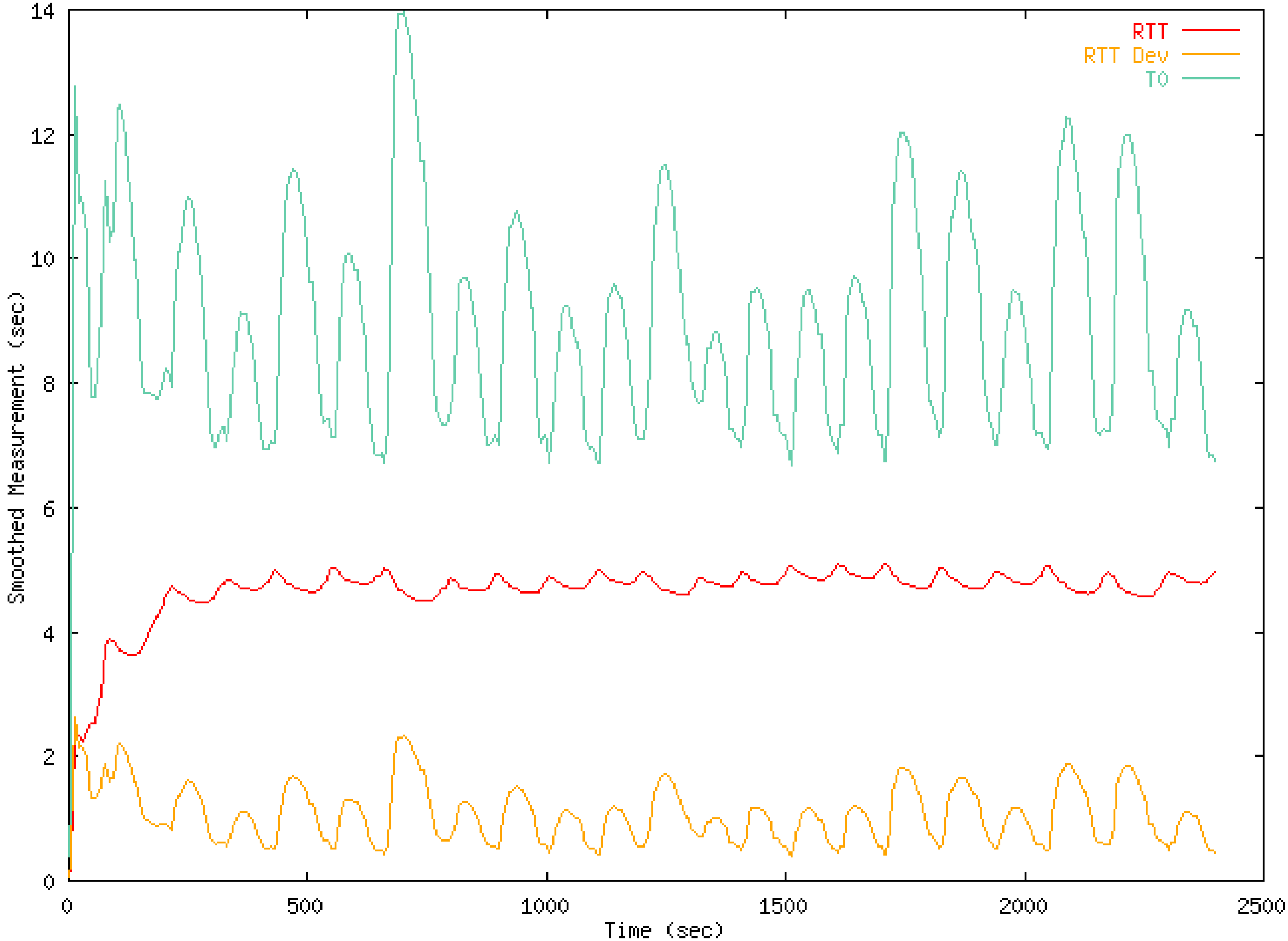


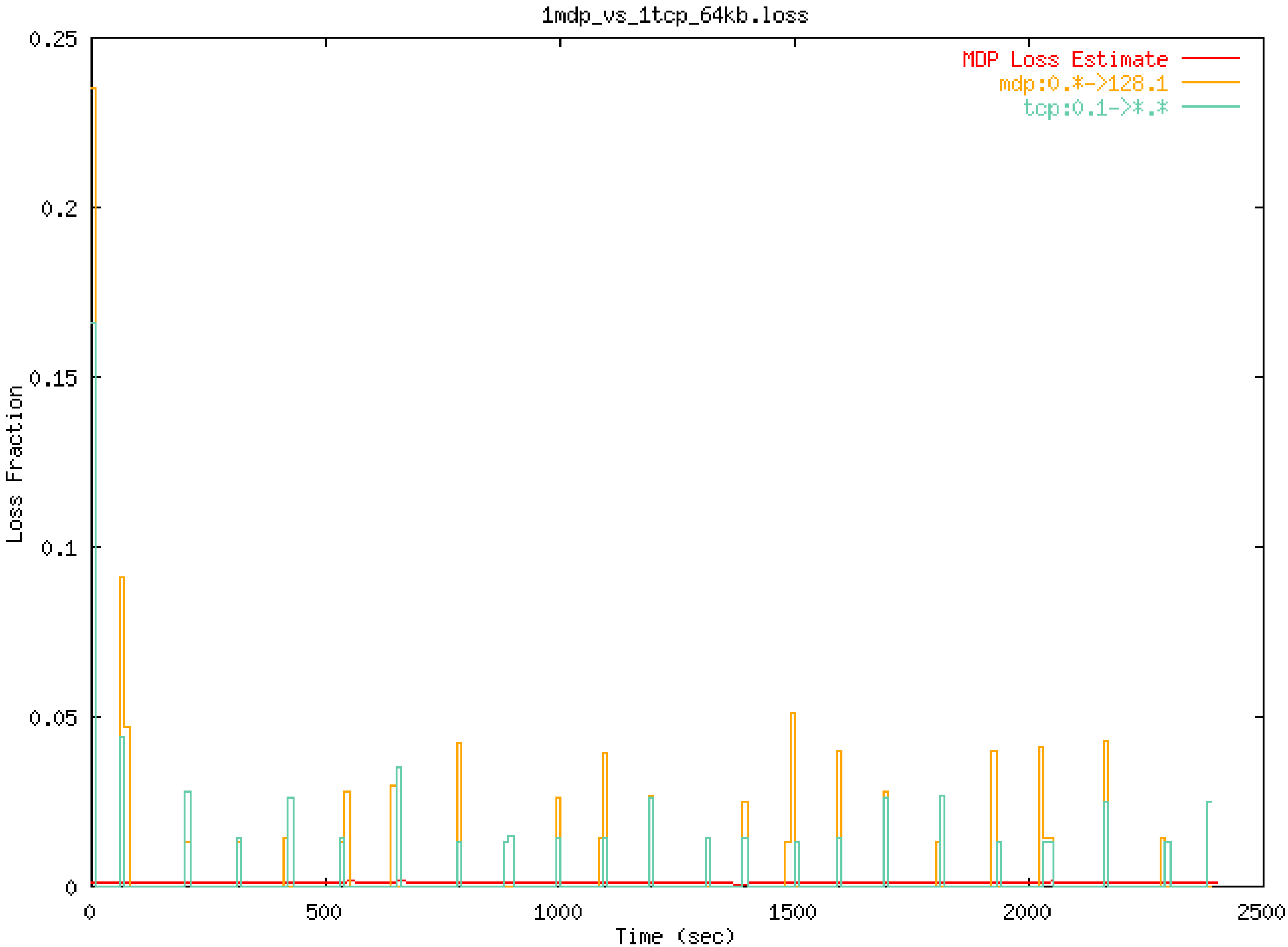
1mdp_vs_1tcp_64kb.rate
Rate (kbps)
Time (sec)
0
10
20
30
40
50
60
70
0
500
1000
1500
2000
2500
mdp:0.*->128.1
tcp:0.1->*.*
:.*->*.*

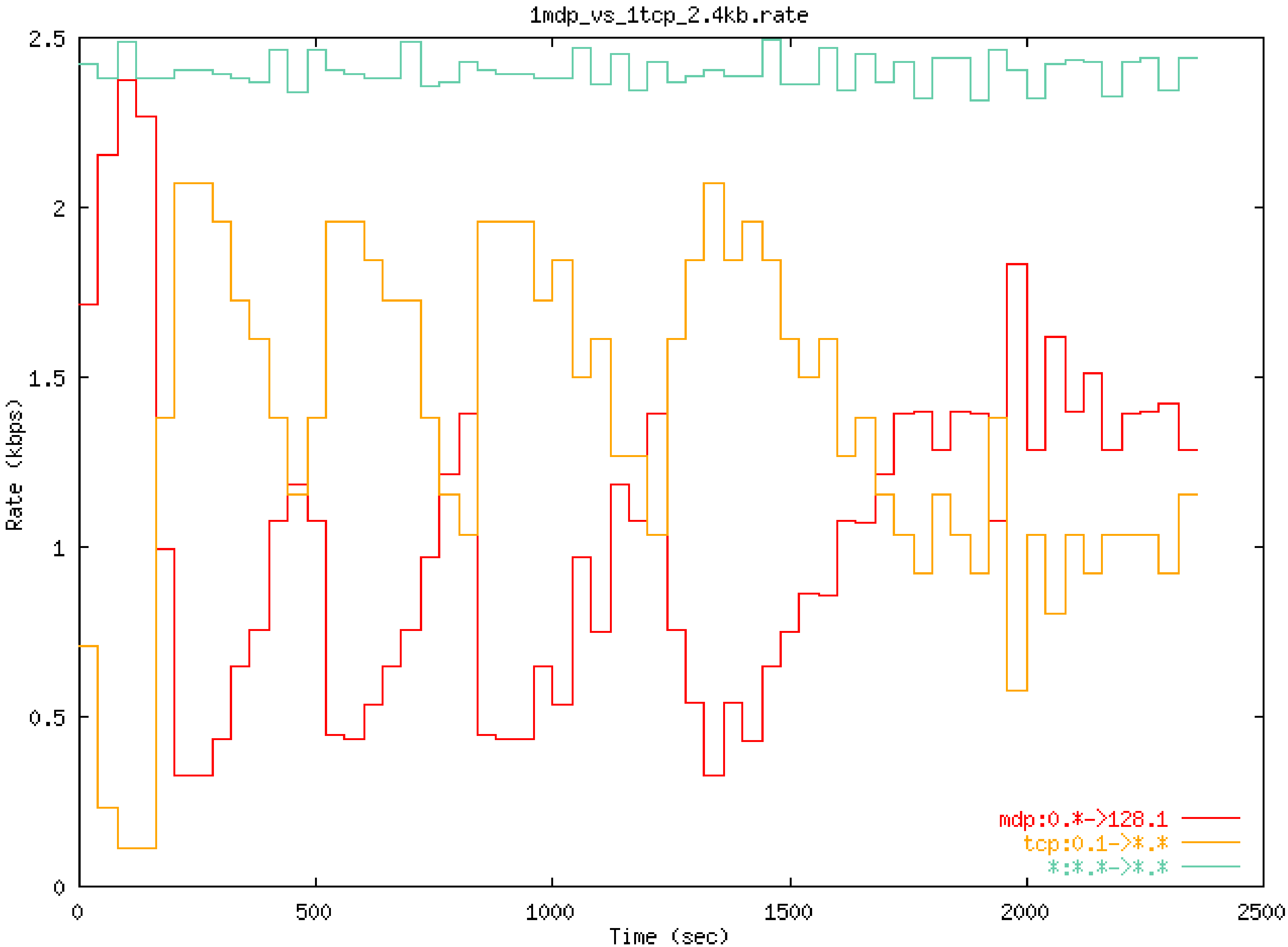
MDP Goal/Tx Rate - 64 kbps Bottleneck



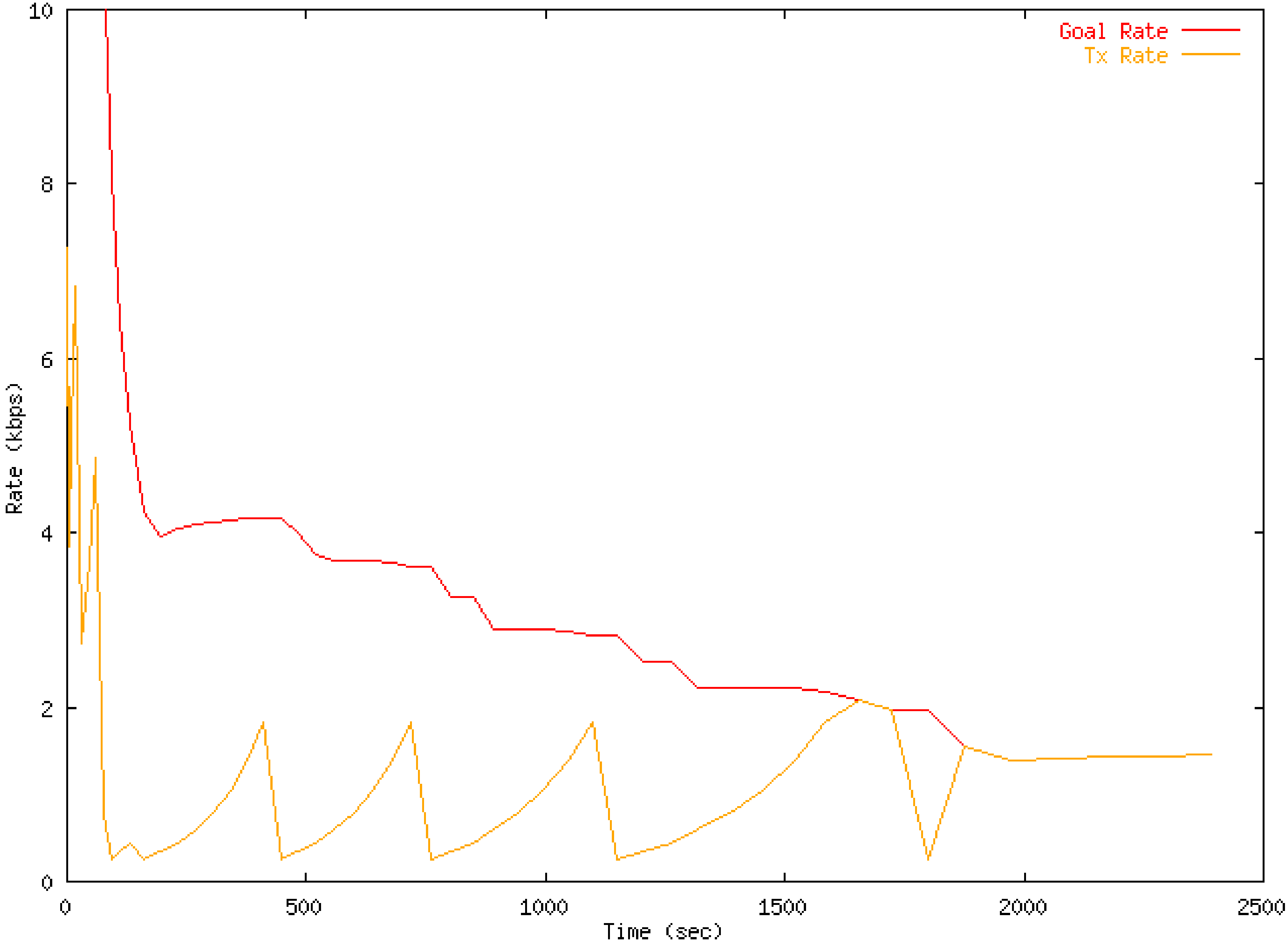
RTT Measurement - 64 kb Bottleneck

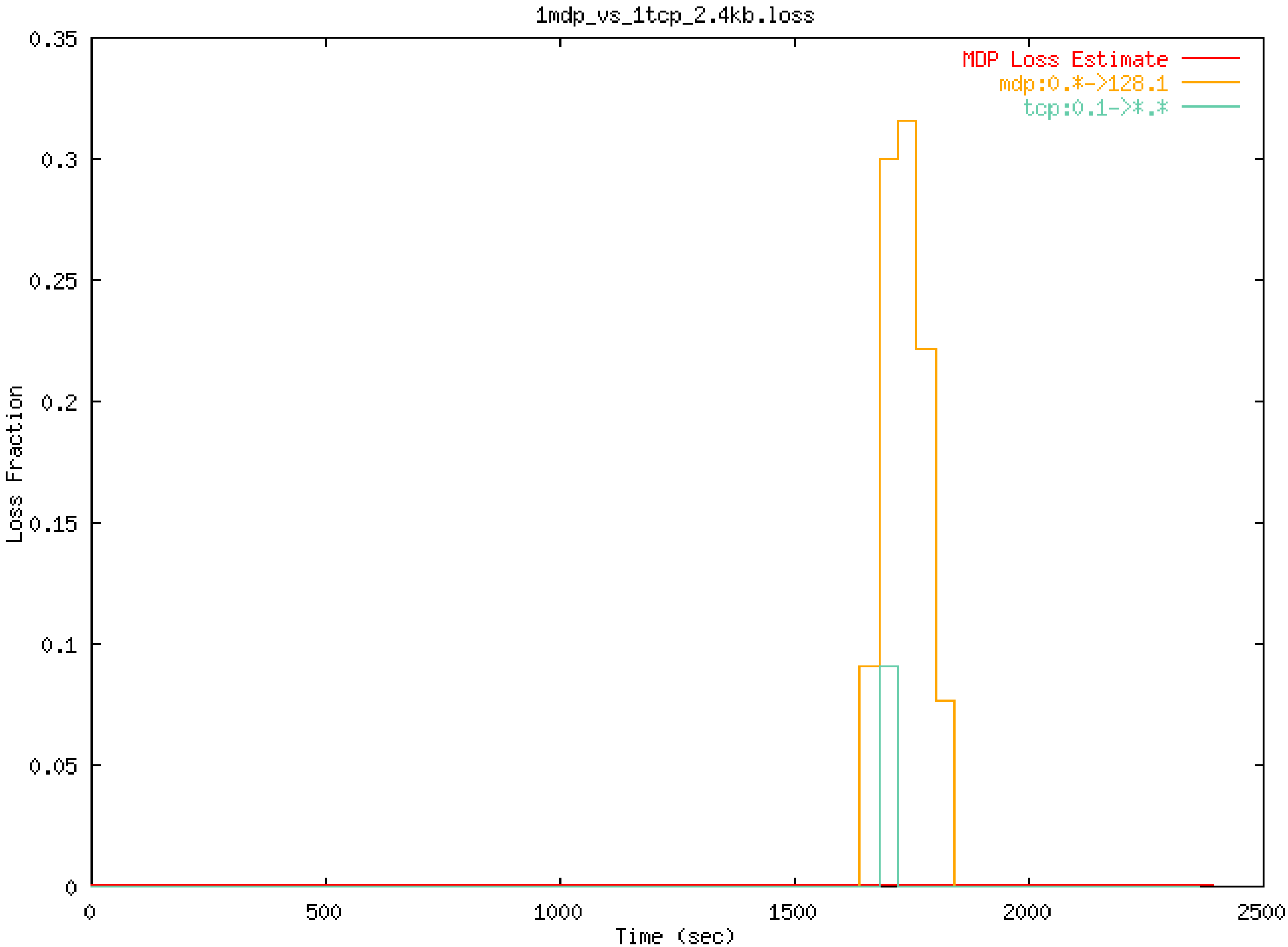



1mdp_vs_1tcp_64kb.loss
MDP Loss Estimate
mdp:0.*->128.1
tcp:0.1->*.*
Loss Fraction
Time (sec)
0
0.05
0.1
0.15
0.2
0.25
500
1000
1500
2000
2500

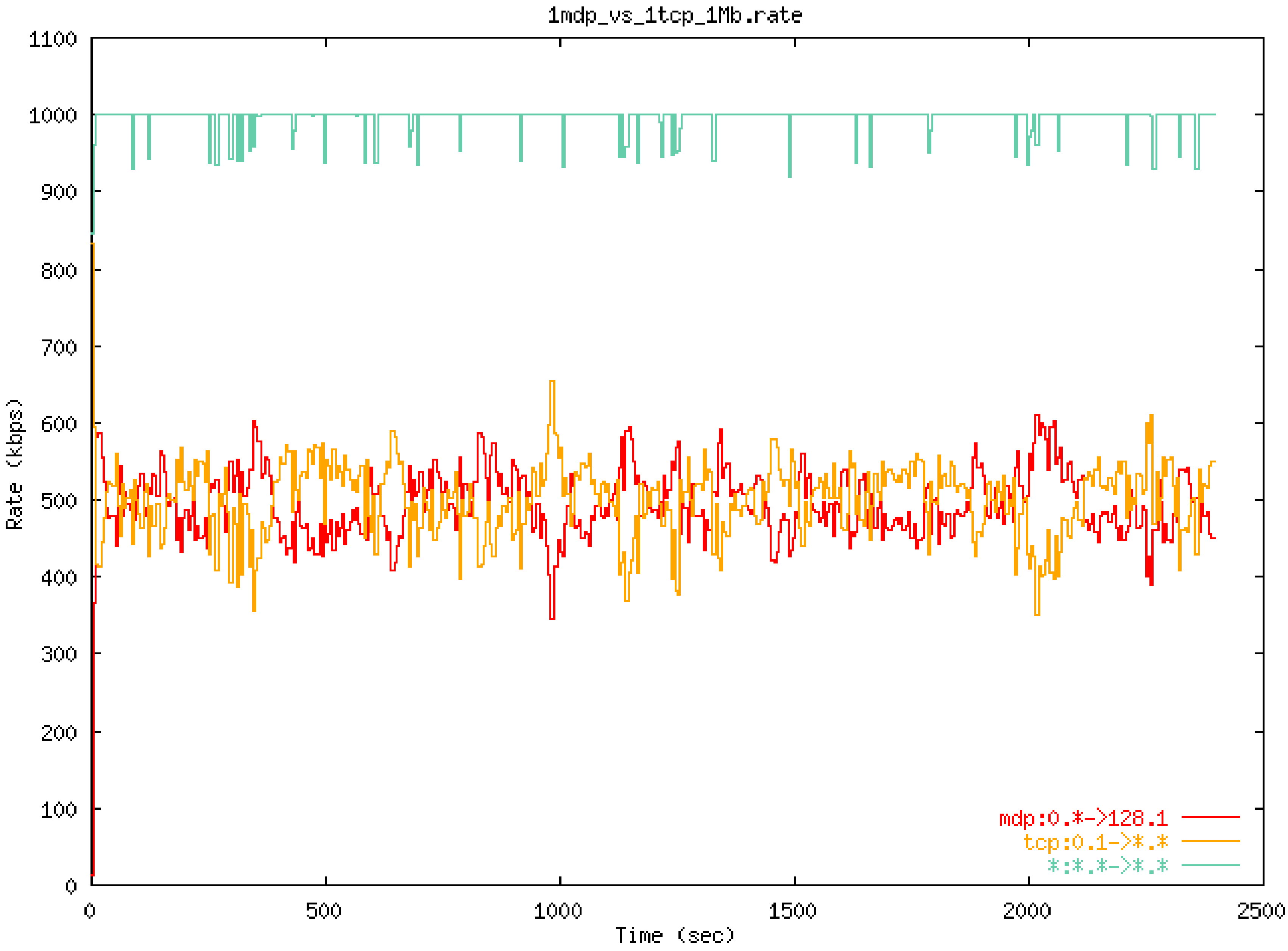

1mdp_vs_1tcp_2.4kb.rate
Rate (kbps)
Time (sec)
0
0.5
1
1.5
2
2.5
0
500
1000
1500
2000
2500
mdp:0.*->128.1
tcp:0.1->*.*
:.*->*.*

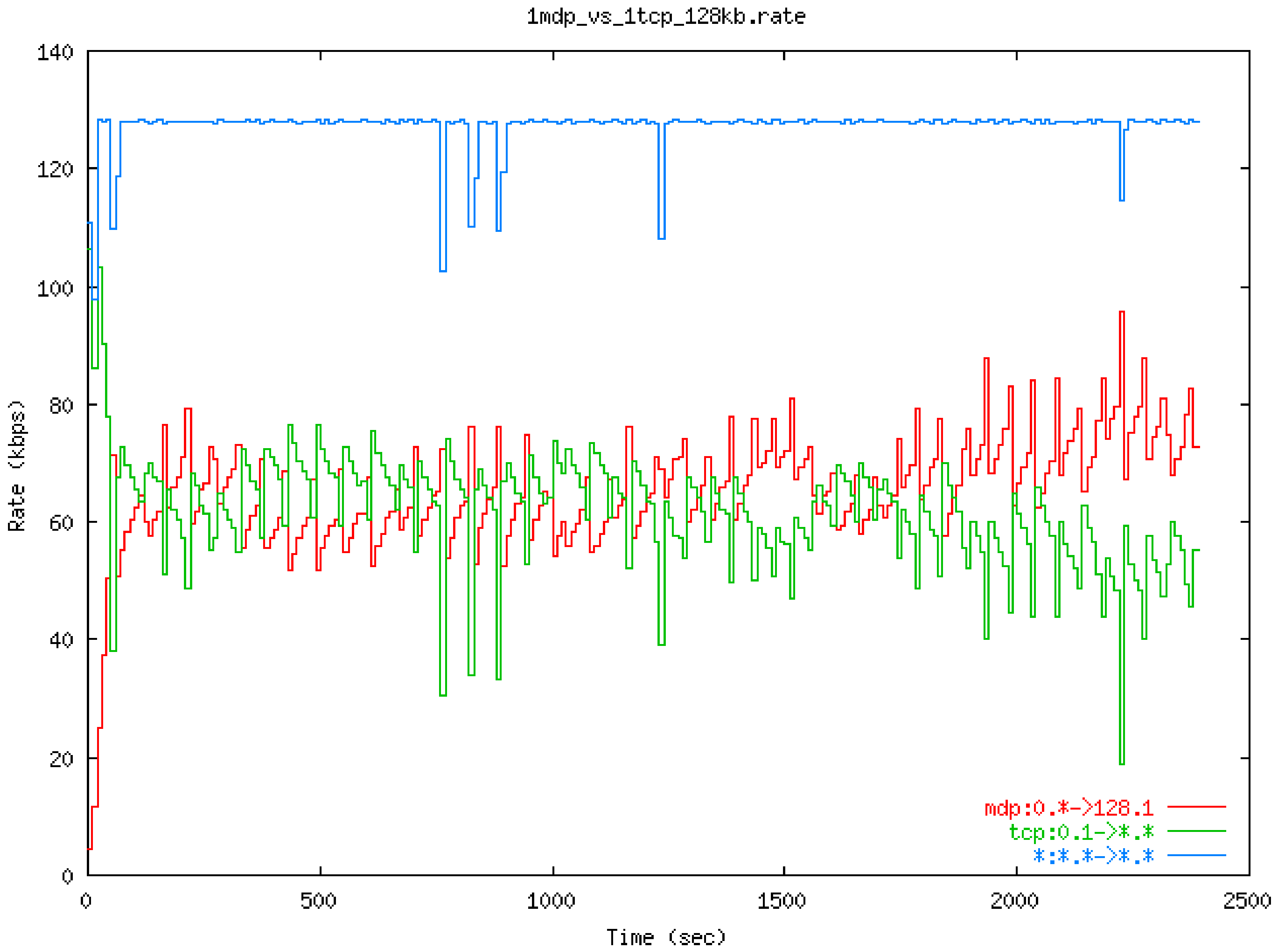
MDP Goal/Tx Rate - 2.4 kbps Bottleneck

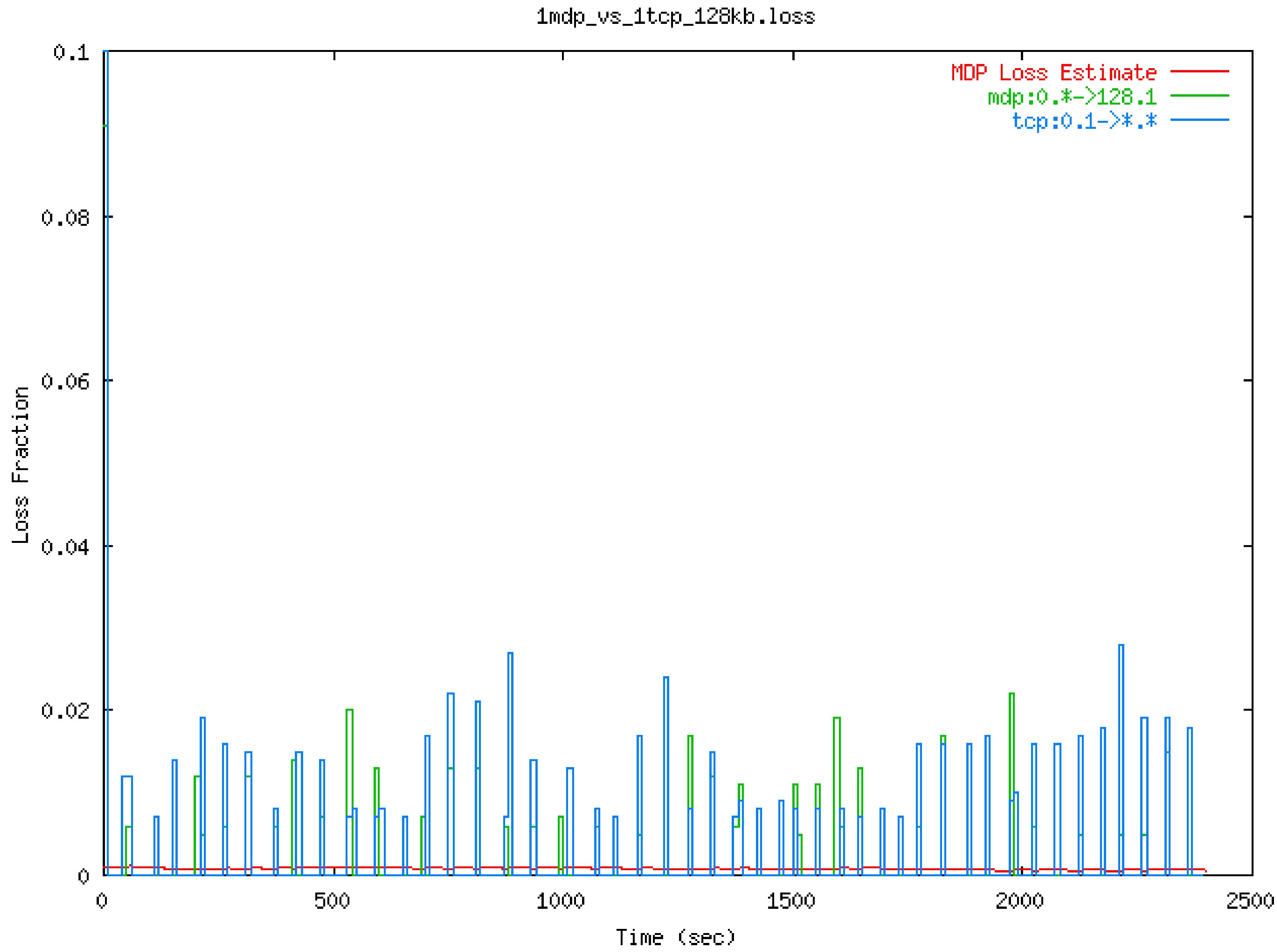


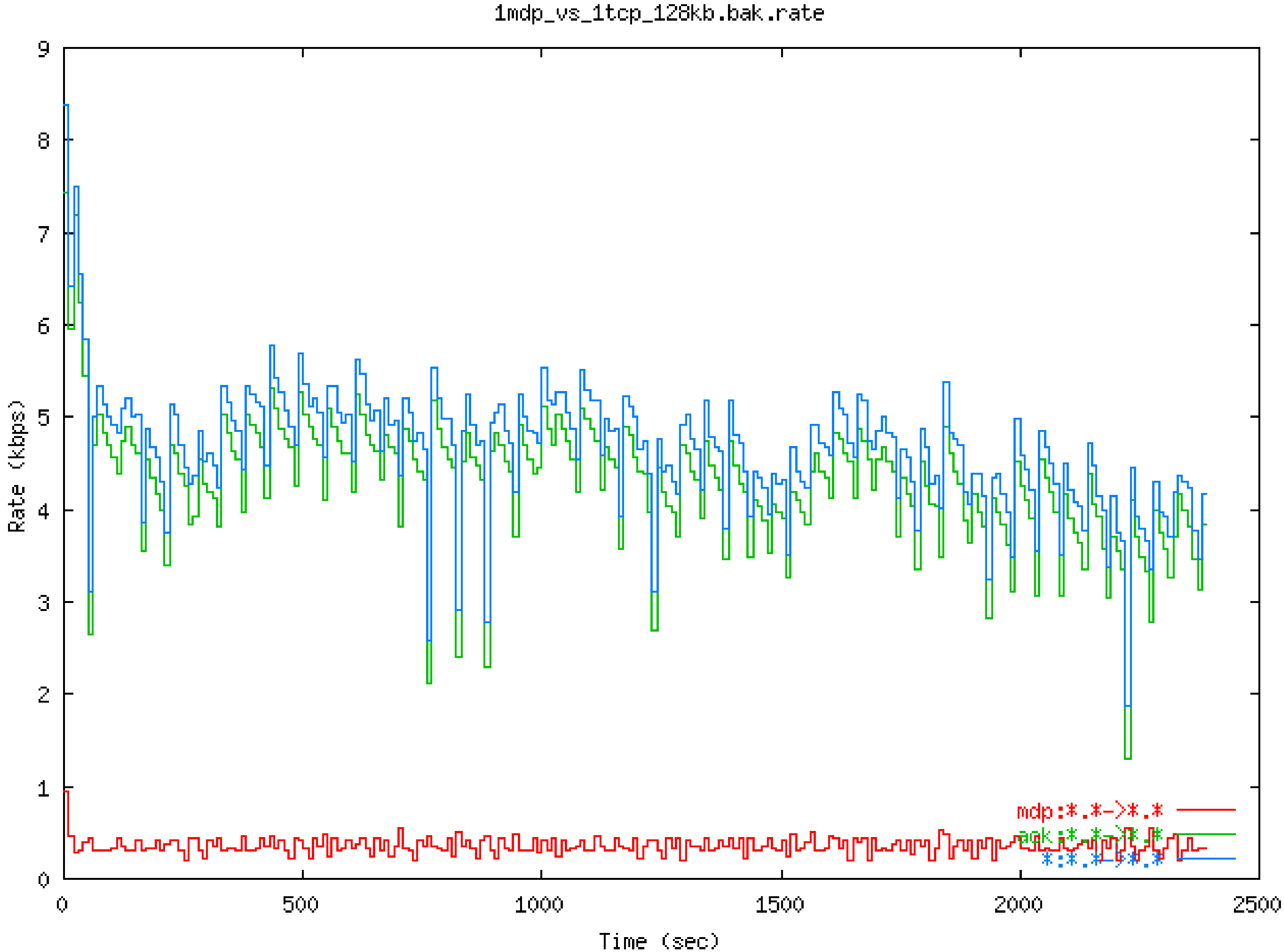


1mdp_vs_1tcp_2.4kb.loss
MDP Loss Estimate
mdp:0.*->128.1
tcp:0.1->*.*
Loss Fraction
Time (sec)
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0
500
1000
1500
2000
2500


1mdp_vs_1tcp_1Mb.rate
Rate (kbps)
Time (sec)
0
100
200
300
400
500
600
700
800
900
1000
1100
0
500
1000
1500
2000
2500
mdp:0.*->128.1
tcp:0.1->*.*
:.*->*.*


1mdp_vs_1tcp_128kb.rate
Rate (kbps)
Time (sec)
0
20
40
60
80
100
120
140
0
500
1000
1500
2000
2500
mdp:0.*->128.1
tcp:0.1->*.*
:.*->*.*


1mdp_vs_1tcp_128kb.loss
MDP Loss Estimate
mdp:0.*->128.1
tcp:0.1->*.*
Loss Fraction
Time (sec)
0.1
0.08
0.06
0.04
0.02
0
0
500
1000
1500
2000
2500


1mdp_vs_1tcp_128kb.bak.rate
Rate (kbps)
Time (sec)
mdp:*.*->*.*
ack:*.*->*.*
:.*->*.*
0
1
2
3
4
5
6
7
8
9
0
500
1000
1500
2000
2500

Some Ongoing Issues

- How dynamic can reliable multicast congestion control be?
- What policies for protocol startup and/or intermittent idle periods should be observed?
- Scalability issues: There are tradeoffs to be made in terms of scalability vs. congestion control dynamic responsiveness. How should these be weighed?
- Flow control vs. congestion control.

Future Directions

- Further evaluation and refinement of approach in richer multicast topologies and in prescence of network dynamics (bursty traffic, HTTP, different TCP algorithms, etc)
- Further investigation/validation of designated receiver set election.
- Scalability enhancements (e.g. group size estimation for enhanced NACK suppression,
- Evaluation of client feedback rate tradeoff. (I.e. How rapidly does congestion control probing need to occur?)
- Flow control considerations
- Hybrid and/or alternative congestion/flow control techniques. (e.g. proactive network feedback in the form of ECN, etc)